Frederik.Beaujean@lmu.de
Excellence cluster universe, LMU Munich

CHEP 2015, Okinawa

Given data from LHC, what are likely values of masses, cross sections...?
Limits including systematic uncertainties?
Bayes’ theorem

Learning rule

\[ P(\theta|D, M) \propto P(D|\theta, M)P_0(\theta|M) \]

posterior \( \propto \) likelihood \( \times \) prior
**Applying Bayes’ theorem**

**Integration**

- **marginalization** \( P(\theta_i|D, M) = \int \prod_{j \neq i} d\theta_j P(\theta|D, M) \)
- **evidence** \( P(D|M) = \int d\theta P(D|\theta, M)P_0(\theta|M) \)
- **quadrature \rightarrow curse of dimensionality**
Applying Bayes’ theorem

Integration

- marginalization \( P(\theta_i|D, M) = \int \prod_{j \neq i} \text{d}\theta_j \ P(\theta|D, M) \)
- evidence \( P(D|M) = \int \text{d}\theta \ P(D|\theta, M)P_0(\theta|M) \)
- quadrature \(\rightarrow\) curse of dimensionality

\(\Rightarrow\) need samples from posterior
Markov chain Monte Carlo

Metropolis Hastings algorithm

- one sample per step
  1. propose move
  2. accept or stay
Markov chain Monte Carlo

Metropolis Hastings algorithm
one sample per step
1. propose move
2. accept or stay

- marginals
- sample near mode $\Rightarrow$ seed for optimization
- uncertainty propagation $f(\theta) \rightarrow P(f|D, M)$
**Bayesian Analysis Toolkit**

- home page [http://mpp.mpg.de/bat](http://mpp.mpg.de/bat)
- fork me on [https://github.com/bat/bat/bat](https://github.com/bat/bat)

**Motivation**

- reinventing the wheel time waster, error prone
- C++ toolkit to supply algorithms/models ⇒ user can focus on problem
Bayesian Analysis Toolkit

- home page http://mpp.mpg.de/bat
- fork me on https://github.com/bat/bat

Features

- implemented: MCMC (multithreaded), simulated annealing . . .
- depends on ROOT: I/O, plots, optimization (Minuit) . . .
- optional: roostats, CUBA (integration)
- docs, tutorials, examples . . . on web page
COMPONENTS

\[ P(\theta|D, M) \propto P(D|\theta, M)P_0(\theta|M) \]

USER DEFINED

- create model
- read data
\[ P(\theta|D, M) \propto P(D|\theta, M)P_0(\theta|M) \]

**USER DEFINED**
- create model
- read data

**DEFINE**

```python
MyModel : BCMModel
- AddParameter("mu", 0, 1)
- LogLikelihood()
- LogAPrioriProbability()
```
\[ P(\theta|D, M) \propto P(D|\theta, M)P_0(\theta|M) \]

**USER DEFINED**
- create model
- read data

**COMMON TOOLS**
- Normalize()
- FindMode()
- MarginalizeAll()
- PrintAllMarginalized()
- PrintKnowledgeUpdatePlots()

**DEFINE**
**MyModel** : **BCMModel**
- AddParameter("mu", 0, 1)
- LogLikelihood()
- LogAPrioriProbability()
// define the model
BCMTF m("SingleChannelMTF");
  m.AddChannel("channel1");
  m.SetData("channel1", hist_data);
  m.AddProcess("background", 200., 400.);
  m.SetTemplate("channel1", "background",
  1.0);
  m.SetPriorGauss("background", 300., 10.);
  m.SetPriorConstant("signal");
// run MCMC, find mode, then plot
m.MarginalizeAll();
m.FindMode(m.GetBestFitParameters());
m.PrintKnowledgeUpdatePlots("upd.pdf");
m.PrintCorrelationPlot("corr.pdf");
**ATLAS: Z’**

**CMS: quantum black hole**
arXiv:1501.04198v2

**Belle: dark photon**
arXiv:1502.00084

**UTFIT: D meson mixing**
arXiv:1402.1664

110 citations @inspire, ~ 100 downloads of v0.9.4.1 since Jan 20, 2015
<table>
<thead>
<tr>
<th>History</th>
</tr>
</thead>
<tbody>
<tr>
<td>• first release 2008</td>
</tr>
<tr>
<td>• subversion</td>
</tr>
</tbody>
</table>

<p>| |</p>
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>• one of two main developers left physics</td>
</tr>
</tbody>
</table>

Frederik Beaujean

Apr 2015

11 / 14
# Lessons in Software Engineering

## History
- first release 2008
- subversion
- one of two main developers left physics

## Present
1. better code with git: distributed, code review
**Lessons in Software Engineering**

### History
- first release 2008
- subversion
- one of two main developers left physics

### Present
1. better code with git: distributed, **code review**
2. benefit from github: discuss issues, fork, **pull requests**...
**Lessons in Software Engineering**

**History**
- first release 2008
- subversion
- one of two main developers left physics

**Present**
1. better code with git: distributed, code review
2. benefit from github: discuss issues, fork, pull requests...
3. write unit tests: refactor code, add features w/o worrying, automatic tests on different platforms

⇒ time investments pay off
**Improvements Under Development**

- **Ease of use**: streamline option setting, building ...
- Factorized priors $P(\theta|M) = \prod_i P(\theta_i|M)$
  - $\Rightarrow$ community extensible
- Sharing samples as ROOT files (even w/o the model)
  - $\Rightarrow$ uncertainty propagation, replotting
- Multivariate proposal $\Rightarrow$ big speed-up in high dimensions
- Evidence from MCMC \[arXiv:1410.7149\]
  - $\Rightarrow$ release in summer 2015
Wishlist for the future

- threads + MPI for tough problems ⇒ rewrite
- interface to script languages: python, mathematica, R ...
- sampling algorithms: MCMC, Hamiltonian MC, nested sampling, variational Bayes + importance sampling ...
1. Bayes: random numbers
2. BAT well established
3. more powerful sampling algorithms in BAT 2.0